(12) UK Patent Application (19) GB (11) 2 369 746 (13) A

(43) Date of A Publication 05.06.2002

(21) Application No 0029179.9

(22) Date of Filing 30.11.2000

(71) Applicant(s)
Ridgeway Systems & Software Ltd
(Incorporated in the United Kingdom)
66 Suttons Business Park, READING, Berkshire, RG6 1AZ, United Kingdom

(72) Inventor(s)
Stephen Michael Read

(74) Agent and/or Address for Service
Dummett Copp
25 The Square, Martlesham Heath, IPSWICH, Suffolk, IP5 3SL, United Kingdom

(51) INT CL^7
H04L 29/12

(52) UK CL (Edition T)
H4K KTKX

(56) Documents Cited
WO 01/30036 A1 **US 6058431 A**

(58) Field of Search
UK CL (Edition S) **H4K KTKX**
INT CL^7 **H04L 29/12**
On-Line - EPODOC, JAPIO, WPI

(54) Abstract Title
Communications system with network address translation

(57) The present invention relates to a communications system 1 for handling communications sessions, for example multimedia calls or voice calls. The communications system comprises a local terminal 10, an external server 40, a proxy interface agent 11 between the terminal 10 and a shared network 20. The communication means includes a NAT function 32 through which the communications session must pass. The communications session is carried over the network over one or more logical channels between the terminal 10 and the external server 40, during which the first NAT function 32 applies network address mappings on the terminal's transport addresses 14. The proxy interface agent 11 acts on behalf of the terminal in communications with the external server and establishes a logical channel on an outbound connection to the server that serves as a control channel between the proxy interface agent and the server. The proxy interface agent establishes dynamic outbound connections to the server, and in response to a request from the server, makes one or more associations between the terminal's transport address(es) and identifiable logical channel(s) between the proxy interface agent and the server. These identifiable logical channel(s) are established on one or more of the dynamic outbound connections from the proxy interface agent to the server.

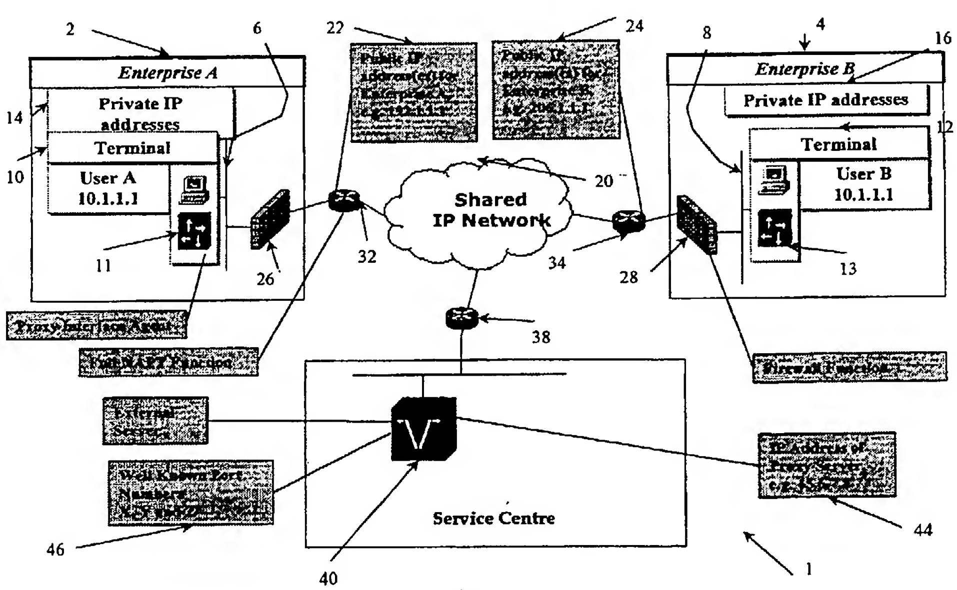

Fig. 1

GB 2 369 746 A

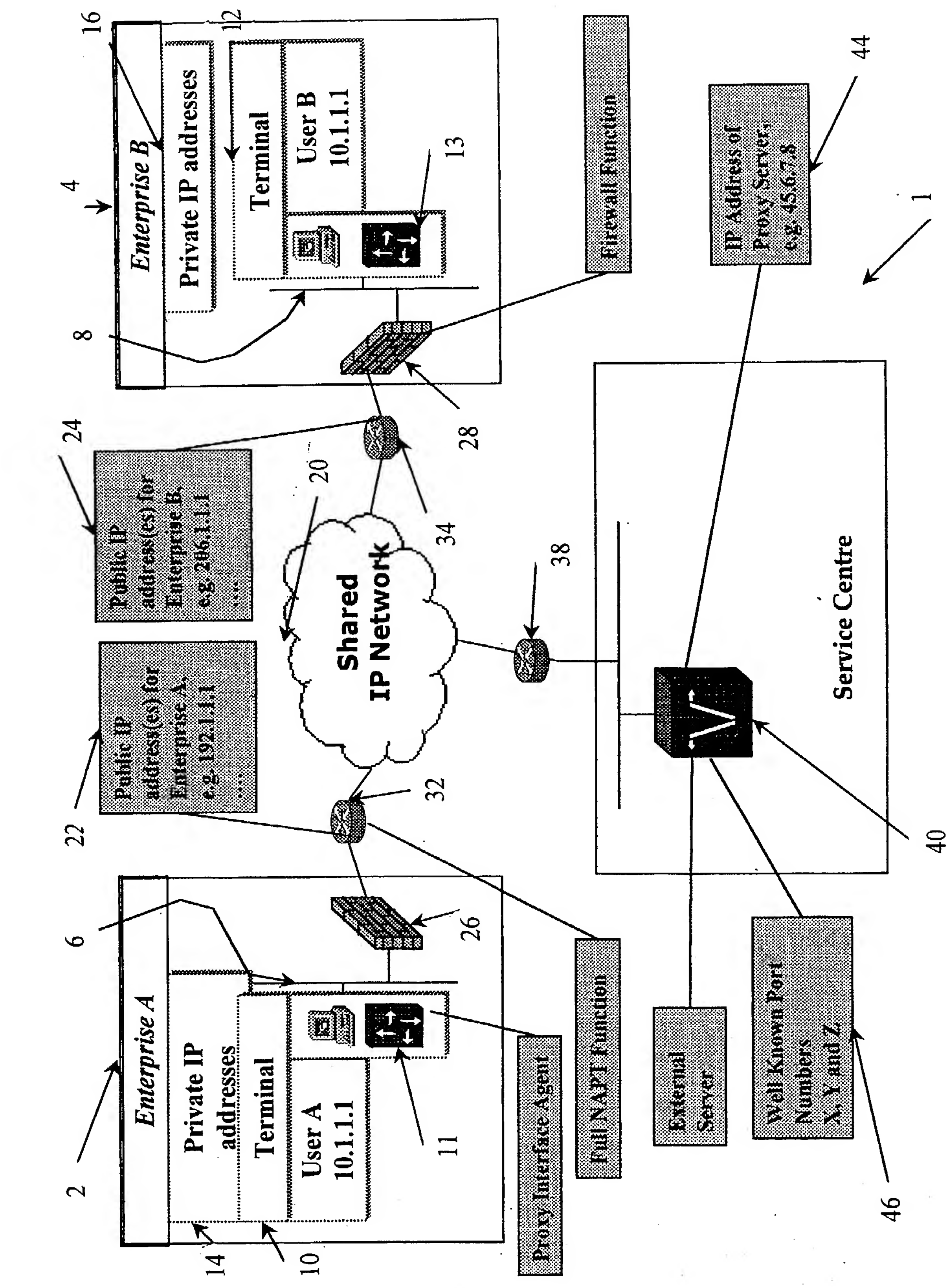
Enterprise A
Private IP addresses
Terminal
User A
10.1.1.1
Enterprise B
Private IP addresses
Terminal
User B
10.1.1.1
Public IP address(es) for Enterprise A, e.g. 192.1.1.1
...
Public IP address(es) for Enterprise B, e.g. 206.1.1.1
...
Shared IP Network
Proxy Interface Agent
Full NAPT Function
Firewall Function
IP Address of Proxy Server, e.g. 45.6.7.8
Service Centre
External Server
Well Known Port Numbers X, Y and Z
1
2
4
6
8
10
11
12
13
14
16
20
22
24
26
28
32
34
38
40
44
46

Fig. 1

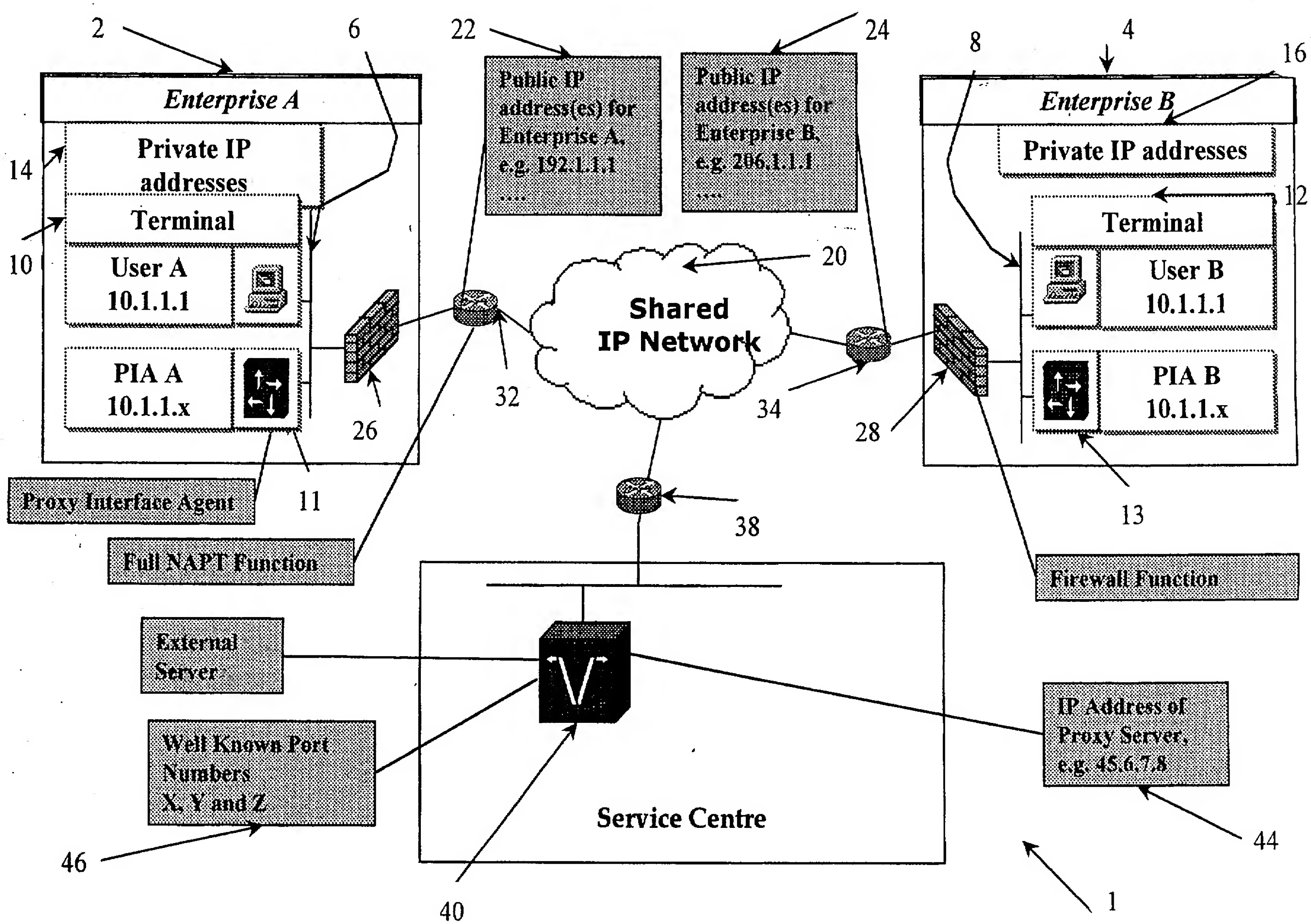
Enterprise A
Private IP addresses
Terminal
User A 10.1.1.1
PIA A 10.1.1.x
Proxy Interface Agent
Full NAPT Function
Public IP address(es) for Enterprise A, e.g. 192.1.1.1 ....
Public IP address(es) for Enterprise B, e.g. 206.1.1.1 ....
Shared IP Network
Enterprise B
Private IP addresses
Terminal
User B 10.1.1.1
PIA B 10.1.1.x
Firewall Function
External Server
Well Known Port Numbers X, Y and Z
Service Centre
IP Address of Proxy Server, e.g. 45.6.7.8
2
6
22
24
4
8
16
14
10
12
20
32
34
26
28
11
13
38
46
40
44
1

Fig. 2

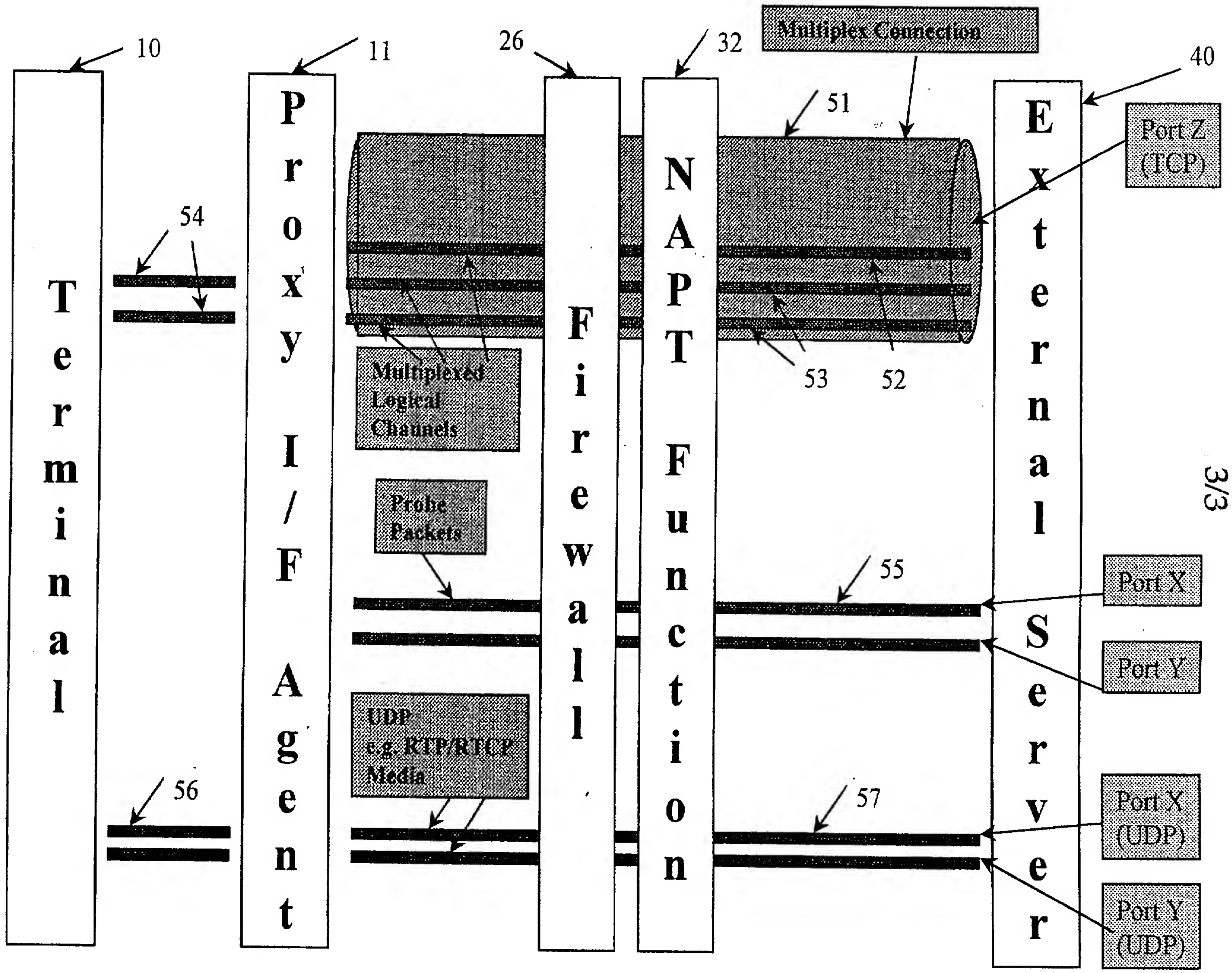
10
11
26
32
40
Terminal
Proxy I/F Agent
Firewall
NAPT Function
External Server
Multiplex Connection
51
52
53
54
55
56
57
Port Z (TCP)
Port X
Port Y
Port X (UDP)
Port Y (UDP)
Multiplexed Logical Channels
Probe Packets
UDP e.g. RTP/RTCP Media

Fig. 3

2369746

Communications System

The present invention relates to a communications system for handling communications sessions, for example multimedia calls or voice calls.

This document presents an invention that allows endpoints (using a real-time protocol, for example H.323, SIP or MGCP) located in different secure and private IP data networks to be able to communicate with each other without compromising the data privacy and data security of the individual private networks. The invention relates to a method and apparatus that has the advantage of working with existing security functions, firewalls for example, and NAPT (Network Address Port Translation) functions that may occur in firewalls, routers and proxies. The benefit of the invention is that it saves on the costs of upgrading those devices to be fully protocol (e.g. H.323) compliant or deploying additional protocol aware (e.g. H.323) devices. The invention presented in this document applies to those deployments where simple (1-to-1) NAT (Network Address translation) mapping may be applied at the edge of the private networks and/or to deployments where NAPT (Network Address and Port Translation) is applied at the edge of the private networks. The 2 configurations can coexist and the apparatus can allow communications to take place between private networks following one configuration and private networks following the other configuration. Similarly within a single private network, some terminals may use one configuration (e.g. dedicated room systems) whereas other terminals may use the second configuration (e.g. desktop client PCs). Note that for the purpose of this document NAT will refer to

all types of network address translation.

The invention presented in this document is illustrated with reference to the ITU H.323 standard as that is the predominant standard for real-time multimedia communications over packet networks including IP networks. However, it is equally applicable to other standards or methods that need to dynamically assign ports to carry bi-directional information (e.g. IETF Session Initiation Protocol (SIP)). It is a major benefit of this invention that the private network infrastructure (firewalls and routers) need not be aware of the protocol used for real-time communication, and that the method of tunnelling real-time traffic in and out of a private network may also be protocol agnostic. This allows enterprises to deploy apparatus without regard to the protocol. That is not to say that some implementations may provide 'protocol' checking for security or other reasons.

The rapidly evolving IP (Internet Protocol) data network is creating new opportunities and challenges for multimedia and voice Communications Service Providers. Unprecedented levels of investment are being made in the data network backbone by incumbent telecommunication operators and next generation carriers and service providers. At the same time, broadband access technologies such as DSL and cable modems are bringing high speed Internet access to a wide community of users. The vision of service providers is to make use of the IP data network to deliver new voice, video and data services right to the desktop, the office and the home alongside high speed Internet access.

The H.323 standard applies to multimedia communications over Packet Based Networks that have no guaranteed quality of service. It has been designed to be independent of the underlying transport network and protocols. Today the IP data network is the default and ubiquitous packet network and the majority (if not all) of implementations of H.323 are over an IP data network. Other protocols for real-time (voice and video) communications, for example, SIP and MGCP also use the IP data network for the transport of call signalling and media. New protocols for new applications associated with the transport of real-time voice and video over IP data networks are also expected to be developed. The methods presented within this invention will also apply to them, and other protocols that require multiple traffic flows per single session.

The importance of standards for wide spread communications is fundamental if terminals from different manufacturers are to inter-operate. In the multimedia arena, the current standard for real-time communications over packet networks (such as IP data networks) is the ITU standard H.323. H.323 is now a relatively mature standard having support from the multimedia communications industry that includes companies such as Microsoft, Cisco and Intel. For example, it is estimated that 75% of PCs have Microsoft's NetMeeting (trade mark) program installed. NetMeeting is an H.323 compliant software application used for multimedia (voice, video and data) communication. Interoperability between equipment from different manufacturers is also now being achieved. Over 120 companies world-wide attended the last interoperability event hosted by the International Multimedia Telecommunications Consortium (IMTC), an independent

organisation that exists to promote the interoperability of multimedia communications equipment. The event is a regular one that allows manufacturers to test and resolve inter-working issues.

Hitherto, there had been a number of barriers to the mass uptake of multimedia (particularly video) communications. Ease of use, quality, cost and communications bandwidth had all hampered growth in the market. Technological advances in video encoding, the ubiquity of cheap IP access and the current investment in the data network coupled with the rollout of DSL together with ISDN and Cable modem now alleviates most of these issues making multimedia communications readily available.

As H.323 was being defined as a standard, it was assumed that there would be H.323-H.320 gateways that exist at the edge of network domains converting H.323 to H.320 for transport over the wide area between private networks. Therefore, implementations of H.323 over IP concentrated on communications within a single network.

However, IP continues to find favour as the wide area protocol. More and more organisations continue to base their entire data networks on IP. High speed Internet access, managed Intranets, Virtual Private Networks (VPNs) all based on IP are commonplace. The IP trend is causing H.320 as a multimedia protocol to decline. The market demand is to replace H.320 completely with H.323 over IP. But perhaps the main market driver for transporting real-time communications over IP across the WAN (wide area network) is voice. With standards such as H.323 and SIP users had begun to use the Internet for cheap voice calls

using their computers. This marked the beginning of a whole new Voice over IP (VoIP) industry that is seeing the development of new VoIP products that include Ethernet telephones, IP PBXs, SoftSwiches and IP/PSTN gateways all geared at seamlessly delivering VoIP between enterprises and users. H.323, SIP and MGCP are expected to be the dominant standards here.

Unfortunately, unforeseen technical barriers to the real-world, wide area deployment of H.323 still exist. The technical barriers relate to the communications infrastructure at the boundaries of IP data networks.

Consequently, today, successful implementation of multimedia or voice communications over IP are confined to Intranets or private managed IP networks.

The problems arise because of two IP technologies - Network Address Translation (NAT) and Firewalls. Security is also an issue when considering solutions to these problems. Where deployments of real-time communications over the data networks transverse shared networks (for example the public Internet), enterprises must be assured that no compromise to their data security is being made. Current solutions to these problems require the outside or external IP address(es) of enterprise to become public to anyone with whom that enterprises wishes to communicate (voice communications usually includes everyone). The invention presented herein does not suffer this shortfall as enterprises external IP address(es) need only be known to the 'trusted' service provider which is how the public Internet has largely evolved.

NAT has been introduced to solve the 'shortage of addresses' problem. Any endpoint or 'host' in an IP network has an 'IP address' to identify that endpoint so that data packets can be correctly sent or routed to it and packets received from it can be identified from where they originate. At the time of defining the IP address field no-one predicted the massive growth in desktop equipment. After a number of years of global IP deployment, it was realised that the number of endpoints wanting to communicate using the IP protocol would exceed the number of unique IP addresses possible from the address field. To increase the address field and make more addresses available requires the entire IP infrastructure to be upgraded. (The industry is planning to do this with IP Version 6 at some point).

The solution of the day is now referred to as NAT. The first NAT solution, which is referred to as simple NAT in IETF RFC1631, uses a one-to-one mapping, came about before the World-Wide Web existed and when only a few hosts (e.g. email server, file transfer server) within an organisation needed to communicate externally to that organisation. NAT allows an enterprise to create a private IP network where each endpoint within that enterprise has an address that is unique only within the enterprise but is not globally unique. These are private IP addresses. This allows each host within an organisation to communicate (i.e. address) any other host within the organisation. For external communication, a public or globally unique IP address is needed. At the edge of the private IP network is a device that is responsible for translating a private IP address to/from a public IP address - the NAT function. The enterprise will have one or more public addresses

belonging exclusively to the enterprise but in general fewer public addresses than hosts are needed either because only a few hosts need to communicate externally or because the number of simultaneous external communications is smaller. A more sophisticated embodiment of NAT has a pool of public IP addresses that are assigned dynamically on a first come first served basis for hosts needing to communicate externally. Fixed network address rules are required in the case where external equipment needs to send unsolicited packets to specific internal equipment.

Today, most private networks use private IP addresses from the 10.x.x.x address range. External communications are usually via a service provider that offers a service via a managed or shared IP network or via the public Internet. At the boundaries between the public and private networks NAT is applied to change addresses to be unique within the IP network the packets are traversing. Simple NAT changes the complete IP address on a one-to-one mapping that may be permanent or dynamically created for the life of the communication session.

Web Servers, Mail Servers and External servers are examples of hosts that would need a static one-to-one NAT mapping to allow external communications to reach them.

A consequence of NAT is that the private IP address of a host is not visible externally. This adds a level of security.

An extension to simple NAT additionally uses ports for the translation mapping and is often referred to as NAPT (Network Address Port Translation) or PAT (Port Address

Translation). A port identifies one end of a point-to-point transport connection between 2 hosts. With mass access to the World-Wide-Web (WWW), the shortage of public IP addresses was again reached because now many desktop machines needed to communicate outside of the private network. The solution as specified in IETF RFC 1631, allows a many-to-one mapping of private IP addresses to public IP address(es) and instead used a unique port assignment (theoretically there are 64k unique ports on each IP address) on the public IP address for each connection made from a private device out into the public or shared network. Because of growth of the Internet, PAT is the common method of address translation.

A peculiarity of PAT is that the private IP address/port mapping to public IP address/port assignments are made dynamically, typically each time a private device makes an outbound connection to the public network. The consequence of PAT is that data cannot travel inbound, that is from the public network to the private network, unless a previous outbound connection has caused such a PAT assignment to exist. Typically, PAT devices do not make the PAT assignments permanent. After a specified 'silence' period has expired, that is when no more inbound data has been received for that outbound initiated connection, the PAT assignment for that connection is unassigned and the port is free to be assigned to a new connection.

While computers and networks connected via a common IP protocol made communications easier, the common protocol also made breaches in privacy and security much easier too. With relatively little computing skill it became possible to access private or confidential data and files

and also to corrupt that business information maliciously. The industry's solution to such attacks is to deploy 'firewalls' at the boundaries of private networks.

Firewalls are designed to restrict or 'filter' the type of IP traffic that may pass between the private and public IP networks. Firewalls can apply restrictions through rules at several levels. Restrictions may be applied at the IP address, the Port, the IP transport protocol (TCP or UDP for example) or the application. Restrictions are not symmetrical. Typically a firewall will be programmed to allow more communications from the private network (inside the firewall) to the public network (outside the firewall) than in the other direction.

It is difficult to apply firewall rules just to IP addresses. Any inside host (i.e. your PC) may want to connect to any outside host (a web server) dotted around the globe. To allow further control the concept of a 'well known port' is applied to the problem. A port identifies one end of a point-to-point transport connection between 2 hosts. A 'well known port' is a port that carries one 'known' type of traffic. IANA, the Internet Assigned Number Authority specifies the well known ports and the type of traffic carried over them. For example port 80 has been assigned for web surfing (http protocol) traffic, port 25 Simple Mail Transport Protocol etc.

An example of a firewall filtering rule for Web Surfing would be:

Any inside IP address/any port number may connect to any outside IP address/Port 80 using TCP (Transport Connection

protocol) and HTTP (the application protocol for Web Surfing).

The connection is bi-directional so traffic may flow back from the Web Server on the same path. The point is that the connection has to be initiated from the inside.

An example of a firewall filtering rule for email may be:

Any outside IP address/any port number may connect to IP address 192.3.4.5/port 25 using TCP and SMTP.

(Coincidentally, the NAT function may change the destination IP address 192.3.4.5 to 10.6.7.8 which is the inside address of the mail server.)

Filtering rules such as: Any inside IP address/any port number may connect to any outside IP address/any port number for TCP or UDP and vice versa are tantamount to removing the firewall and using a direct connection as it is too broad a filter. Such rules are frowned upon by IT managers.

H.323 has been designed to be independent of the underlying network and transport protocols. Nevertheless, implementation of H.323 in an IP network is possible with the following mapping of the main concepts:

H.323 address	:	IP address
H.323 logical channel	:	TCP/UDP Port connection

In the implementation of H.323 over IP, H.323 protocol messages are sent as the payload in IP packets using

either TCP or UDP transport protocols. Many of the H.323 messages contain the H.323 address of the originating endpoint or the destination endpoint or both endpoints. Other signalling protocols such as SIP also embeds IP addresses within the signalling protocol payload.

However, a problem arises in that NAT functions will change the apparent IP addresses (and ports) of the source and destination hosts without changing the H.323 addresses in the H.323 payload. As the hosts use the H.323 addresses and ports exchanged in the H.323 payload to associate the various received data packets with the call, this causes the H.323 protocol to break and requires intermediary intelligence to manipulate H.323 payload addresses.

Because of the complexity of multimedia communications, H.323 requires several logical channels to be opened between the endpoint. Logical channels are needed for call control, capabilities exchange, audio, video and data. In a simple point-to-point H.323 multimedia session involving just audio and video, at least 6 logical channels are needed. In the IP implementation of H.323, logical channels are mapped to TCP or UDP port connections, many of which are assigned dynamically.

As the firewall functions filter out traffic on ports that they have no rules for, either the firewall is opened, which defeats the purpose of the firewall, or much of the H.323 traffic will not pass through.

Therefore, both NAT and firewall functions between endpoints prevent H.323 (and other real-time protocols, SIP and MGCP for example) communications working. This

will typically be the case when the endpoints are in different private networks, when one endpoint is in a private network and the other endpoint is in the Internet or when the endpoints are in different managed IP networks.

H.323 (and SIP, MGCP etc.) communication is therefore an anathema to firewalls. Either a firewall must become H.323 aware or some intermediary intelligence must manipulate the port assignments in a secure manner.

One possible solution to this problem would be a complete IP H.323 infrastructure upgrade. This requires:

- H.323 upgrade to the NAT function at each IP network boundary. The NAT function must scan all H.323 payloads and consistently change IP addresses.
- H.323 upgrade to the firewall function at each IP network boundary. The firewall must understand and watch all H.323 communication so that it can open up the ports that are dynamically assigned and must filter all non-H.323 traffic on those ports.
- Deployment of H.323 intelligence at the boundary or in the shared IP network to resolve and arbitrate addresses. IP addresses are rarely used directly by users. In practice, IP address aliases are used. Intelligence is needed to resolve aliases to an IP address. This H.323 function is contained within H.323 entities called Gatekeepers.

The disadvantages of this possible solution are:

- Each organisation/private network must have the same level of upgrade for H.323 communication to exist.

- The upgrade is costly. New functionality or new equipment must be purchased, planned and deployed. IT managers must learn about H.323.
- The scale of such a deployment will likely not be readily adaptable to the demands placed on it as the technology is progressively adopted, requiring a larger and more costly initial deployment than initial (perhaps experimental) demand requires.
- The continual parsing of H.323 packets to resolve the simple NAT and firewall function places a latency burden on the signal at each network boundary. The latency tolerance for audio and video is very small.
- Because there are a multitude of standards for real-time communication and each of the signalling protocols of those standards are different, an enterprise would need multiple upgrades - one for each protocol it wishes to use.
- The media is expected to travel directly between enterprises or between an enterprise and a device in the public network. The consequence of this is that the IP addresses of an enterprise becomes public knowledge. This is regarded as a security compromise as any potential attacker must first discover the enterprises IP address as the first step to launching an attack.

As a result of these problems, the H.323 protocol is not being used for multimedia communications when there is a firewall and/or network address translation (NAT). One approach has been to place H.323 systems on the public side of the firewall and NAT functions. This allows them to use H.323 while also allowing them to protect the remainder of their network. The disadvantages of this are:

1. The most ubiquitous device for video communications is the desktop PC. It is nonsensical to place all desktop computers on the public side!
2. The H.323 systems are not protected from attackers on the public side of the firewall.
3. The companies are not able to take advantage of the potentially ubiquitous nature of H.323, since only the special systems will be allowed to conduct H.323 communications.
4. The companies will not be able to take full advantage of the data-sharing facilities in H.323 because the firewall will prevent the H.323 systems from accessing the data. Opening the firewall to allow data-transfer functions from the H.323 system is not an option because it would allow an attacker to use the H.323 system as a relay.
5. In the emerging Voice over IP (VoIP) market there is a market for telephony devices that connect directly to the data network, for example Ethernet telephones and IP PBXes. By virtue of the desktop nature they are typically deployed on the private network behind firewalls and NAT. Without solutions to the problems described above telephony using these devices is confined to the Enterprises private network or Intranet or must pass through IP-PSTN gateways to reach the outside world.

The advantages of using the broadband connection to the enterprise for voice and video as well as data requires secure solutions to these issues.

It is an object of the present invention to address these

.problems.

Accordingly, the invention provides a communications system for handling a communications session, comprising a first local terminal, an external server, one or more logical channels between the first local terminal and the external server for carrying the communications session over a shared communications network, said communication means including a first NAT function through which the communications session must pass, in which:

i) the first local terminal has at least one transport address for the communications session;

ii) the first NAT function applies network address mappings on the transport addresses on connections between the first terminal and the shared communications network;

iii) the system includes a first proxy interface agent arranged to act on behalf of the first local terminal in communications with the external server;

iv) the first proxy interface agent is capable of establishing a logical channel on an outbound connection to the external server, said logical channel serving as a control channel between the first proxy interface agent and the external server;

characterised in that:

v) said outbound connections are dynamic outbound connections established by the first proxy interface agent;

vi) the external server is adapted to request the first proxy interface agent to make association(s) between the transport address(es) of the first local terminal and identifiable logical channel(s) between the first proxy interface agent and the external server, said identifiable logical channel(s) being established on one or more of said dynamic outbound connections from the first proxy interface agent to the external server.

The sum of the logical channels provides the communications session and the outbound connections create the necessary NAT mappings that enable inbound and outbound communications between the terminal and the external server. Communication to and from the first local terminal is transparently mapped by the first proxy interface agent onto the identifiable logical channels. The communications system therefore can be used to provide a transparent communications means between the first terminal and the external server.

In order to allow inbound communications over TCP, previously established bi-directional outbound connections are made to establish NAT mappings.

In order to allow inbound communications over UDP, probe packet(s) are sent to establish the NAT mappings.

During the communications session, the first NAT function continues to apply network address mappings to connections between the first proxy interface agent and the external server.

Identifiable logical channels may be multiplexed into one or more connections using normal multiplexing techniques.

An example of a transport address is an IP address plus a port number. The network address mappings will in general therefore be mappings of IP addresses and/or ports.

Also according to the invention, there is provided a method of handling a communications session in a communications system, the communications system comprising a first local terminal, an external server, a first proxy interface agent between the first local terminal and the shared network, said communication means including a first NAT function through which the communications session must pass, in which the method comprises the steps of:

i) carrying the communications session over a shared communications network over one or more logical channels between the first local terminal and the external server, the first local terminal having at least one transport address for the communications session;

ii) allowing the first NAT function to continue to apply network address mappings on the transport addresses on connections between the first terminal and the shared communications network;

iii) using the first proxy interface agent to act on behalf of the first local terminal in communications with the external server;

iv) using the first proxy interface agent to establish a

logical channel on an outbound connection to the external server, said logical channel serving as a control channel between the first proxy interface agent and the external server;

characterised in that the method comprises the steps of:

v) using the first proxy interface agent to establish dynamic outbound connections to the external server;

vi) in response to a request from the external server directed to the first proxy interface agent, using the first proxy interface agent to make one or more associations between the transport address(es) of the first local terminal and identifiable logical channel(s) between the first proxy interface agent and the external server, said identifiable logical channel(s) being established on one or more of said dynamic outbound connections from the first proxy interface agent to the external server.

The transport address(es) of the first local terminal are preferably assigned dynamically. Similarly, the transport address(es) of the external server may be assigned dynamically.

Alternatively, none of the transport address(es) of the external server may be assigned dynamically.

The communications system may include a first firewall through which the communications session must pass. The first firewall is then configured to restrict certain types of communication between the first local terminal

and the shared communications network and being configured not to restrict communication between the first proxy interface agent and the external server.

At least one of the transport address(es) of the external server may have at least one pre-assigned (sometime referred to as 'well-known') port. The outbound connections from the first proxy interface agent to the external server then use said pre-assigned port(s).

Preferably, all the transport address(es) of the external server have pre-assigned ports. In this case, it may be that all the transport address(es) of the external server have at most two pre-assigned ports.

The number of pre-assigned ports of the external server may be less than or equal to the total number of dynamically assigned ports for the terminal(s). For example, the external server may have three pre-assigned ports, one for TCP and two for UDP.

The communications system may include a second local terminal and the external server is a proxy server between the first terminal and the second terminal that acts for each terminal as a proxy for the other terminal during the course of the communications session.

In many cases, there may be a second local terminal with a second firewall and/or second NAT function through which the communication session must pass. The second firewall may then be configured to restrict certain types of communication between the second terminal and the public communications network. The proxy server will then have

logical communication ports for communication with the terminals including, for example, one or more pre-assigned ports for communication with the second terminal. The second firewall can then be configured not to restrict communication between the second terminal and the pre-assigned port(s) of the proxy server, and a second proxy interface agent is deployed to act on behalf of the second terminal in its communication with the proxy server. The second local terminal may then engage in a communications session with a second proxy interface agent in a similar manner to that described above.

The shared communications network will in general include the public communications network and/or the Internet.

The proxy interface agent may be co-located with the local terminal, or alternatively, the proxy interface agent may be remote from the local terminal.

The invention may also be useful in cases where there is more than one local terminal per proxy interface agent. The proxy interface agent can then act simultaneously on behalf of terminals using different real-time protocols, for example both H.323 and SIP. The signalling gateway functionality (for example between H.323 and SIP) is preferably provided within either the external server or the proxy interface agent. Additional features and functionality (for example QOS and/or security via encryption) may be provided by the proxy interface agent and external server transparently to the endpoints.

Such a system may be used for making a voice or a multimedia call according to the H.323 standard of the

International Telecommunications Union. Alternatively, the system may be used for making a voice or a multimedia call according to the SIP standard of the Internet Engineering Task Force. Furthermore, the communications system may support mixed protocol environments.

The proxy interface agent may be co-located with an endpoint (for example a PC terminal) or may reside in a separate device from the endpoint(s) it is acting on behalf of.

The terminals may be adapted to transmit and/or receive multimedia media signals together with associated multimedia control signals, the control signals being sent to one of the pre-assigned ports and the media signals being sent to the other pre-assigned ports.

Preferably, at least one the logical communications ports is a pre-assigned port, said request being sent to the pre-assigned port as an initial request to initiate a communication session.

The communication means may be adapted for making a voice or a multimedia call at least in part via the internet, in which case the external server will have a public internet protocol address by which one or both of the terminals communicate with the external server, the firewall(s) being configured not to restrict communication between the terminal(s) and the pre-assigned port(s)of the external server.

The invention is applicable to the case where there is one or more pair(s) of first terminals and of second

terminals. For example, several first voice or multimedia terminals at one site may each connect to corresponding other second voice or multimedia terminals at a variety of other sites.

The invention allows two terminals located in separate private networks to communicate via a common public (or shared) network in which one or both private networks are connected to the public network via firewalls and/or NATs that restrict certain types of communication. Equally, the invention allows one terminal in a private network to communicate with a terminal in a public network, wherein the two networks are connected by firewalls and/or NATs that restrict certain types of communication.

The invention will be described by reference only to the operation between a first endpoint, herein referred to as the first local terminal and an intermediary server, herein referred to as the external server. The operation between a second terminal and the external server mirrors the operation between the first terminal and the external server. Additionally, where the second terminal is directly connected to the public network, this is equivalent to it being connected to a private network in which the firewall and NAT implement null functions. That is, the firewall does not restrict any connections and the NAT uses the same address on both sides for a given connection.

The invention involves the deployment of an external server in the shared or public network and a proxy interface agent in the private network. The external server may be owned and operated by a public service

provider, and thus will typically already be provisioned prior to an enterprise wishing to deploy H.323 communications across the private/public network boundary. The proxy interface agent may be implemented as part of the terminal, or it may be independent of the terminal implementation, but operate on the same device as the terminal, or it may be installed on a separate device.

When enabled, the proxy interface agent will establish a TCP connection to the external server. This connection will be via the firewall and/or NAT if either or both are present. This requires the firewall to allow outgoing TCP connections to the external server's address & well-known port(s). The NAT is able to provide a private to public address mapping (and vice versa) because the connection is created in the outbound direction. As part of the setup process, the external server may authenticate itself to the proxy interface agent, and the connection may be encrypted. The protocol that operates over this connection allows the multiplexing of multiple signalling protocols. Such signalling protocols include, but are not limited to, H.225 RAS, H.225 call signalling, H.245 and SIP. Indeed, this connection is sufficient for all communications between the first local terminal and external server for which the performance characteristics of a TCP connection are acceptable. Once established, the multiplexed connection will remain largely dormant except for periodic registration messages until an outgoing or incoming call attempt is made. For additional security, this connection may be continually setup and disconnected at regular (short) intervals. Each setup of the connection can potentially create a different port assignment in the NAT function, and new encryption keys. Attackers' chances of

exploiting this connection are consequently reduced.

The transport characteristics of the multiplex connection are, however, not appropriate for real-time media such as audio and video. These require UDP based RTP/RTCP connections to be established between the proxy interface agent and the external server. Both in-bound and out-bound RTP/RTCP connections require UDP traffic in both directions. To send media from the terminal to the public network via the external server, the external server sends H.323 messages to the terminal (via the proxy interface agent using the multiplexed connection) that instruct the terminal to send its media to the proxy interface agent. (This can be done using standard H.323 procedures by populating the various data fields of the H.323 messages with address and port values that give the illusion that the terminal and proxy interface agent are the two ends of the H.323 call.) The proxy interface agent must then establish UDP data exchange both to and from the external server through the firewall and/or NAT.

In principle the proxy interface agent can establish a UDP connection to the external server by simply sending a UDP packet to the address and well-known port(s) of the external server. The firewall can be configured to let this traffic through, and the NAT can create a private-to-public address mapping because the connection is created in the outbound direction. However, a device that handles multiple calls involving many UDP connections (such as the external server) typically uses the IP destination address and port, and/or IP source address and port to associate the UDP information with the appropriate call. In the case of the external server, all the UDP data must be sent to

the same IP address and one of the well-known ports in order to be allowed through the firewall. Therefore the IP destination address and port may not be used to differentiate the various UDP connections. Also, from the perspective of the external server, the NAT will assign an effectively random source IP address and port to the UDP packets that it sends. The result is that the IP source address and ports of the UDP data that arrives at the external server will not correspond to any of the media channels that the external server has negotiated through the various signalling channels.

To solve the association problem, the external server instructs the proxy interface agent (via the TCP based multiplexed connection) to send it a probe packet using the same IP source and destination addresses and ports that the proxy interface agent will send subsequent UDP data for this connection. The probe packet contains a unique token chosen by the external server that allows the external server to associate the received probe packet with the appropriate UDP connection. In turn, the external server can associate the IP source and destination addresses and ports of the probe packet with the UDP connection. Knowing this address and port information the external server can associate UDP data subsequently received with these IP addresses and ports with the appropriate call. In an alternative embodiment of the invention, the token information can be multiplexed in with each UDP packet that is sent. Additionally, multiple logic channels can be multiplex onto the same UDP connection. The advantage of taking the latter approach is to conserve port usage in the proxy interface agent. When a smaller number of TCP and UDP connections are used

because of the multiplexing of logical channels, those connections may be place onto pre-assigned or well-known ports at the proxy interface agent. This allows a further tightening of the firewall rules.

To send data from the external server to the proxy interface agent, it is necessary for a public-to-private address mapping to be made in the NAT. As this is typically a 1-to-many mapping, NATs are typically unable to dynamically make such a mapping. However, it is observed that the network path established when making an outgoing UDP connection from proxy interface agent to external server as described above is in actual fact bi-directional in nature. Hence, to establish a UDP connection from the external server to the proxy interface agent, the same steps as for establishing a UDP connection from the proxy interface agent to the external server are followed. However, once the association of addresses and ports is established, the external server uses this information to send UDP data rather than receive UDP data. The proxy interface agent will then send the UDP data on to the terminal. Standard H.323 signalling using appropriate address and port values can be used to prepare the terminal to receive the UDP data from the proxy interface agent.

The invention will be described by way of example, with reference to the accompanying drawings, in which:

Figure 1 is a schematic diagram of a communications system according to the invention for making a voice or a multimedia call between two enterprises in which the proxy interface agent is co-located with an endpoint;

Figure 2 is a schematic diagram similar to that of Figure 1, except that the proxy interface agent is remote from the endpoint; and

Figure 3 is a schematic diagram of the communications systems of Figures 1 and 2, showing the logical channels on outbound connections for both outbound and inbound communications, at one enterprise between the local terminal and the external server.

The alternative to a complete H.323 upgrade is presented in the example described with reference to Figure 1. This shows a communication system 1 having a first enterprise 2 and a second enterprise 4, each of which include private networks 6,8 both of which have one or more H.323 terminals 10,12. Each private network 6,8 has private IP addresses 14,16 coincidentally within the 10.x.x.x address range. The private IP addresses 14,16 may result from a static assignment or dynamic assignment through normal DHCP procedures. Included in the private networks 6,8 are proxy interface agents 11,13 that act on behalf of terminals 10,12 respectively. If the proxy interface agents are not co-located with their respective terminal(s), then the proxy interface agent(s) will have a

unique IP address within the range of their respective private networks 14,16. In such cases, each proxy interface agent 11,13 may act on behalf of multiple terminals 10,12. In figure 1, the proxy interface agents are shown as co-located, and in figure 2 they are shown not co-located. External communication is via a shared, managed or public Internet 20. For external communication, the first enterprise 2 has one or more public IP address(es) 22, for example in a range beginning at 192.1.1.1 and the second enterprise 4 has one or more public IP address(es) 24, for example in a range beginning at 206.1.1.1. Each enterprise has a router 32,34 that applies Network Address Port Translation (NAPT) to dynamically map between inside IP addresses 14,16 and port numbers on those addresses(private) and one of the outside IP addresses 22,24 and the port numbers on the select IP address(public).

The private networks 6,8 are optionally each protected at their edges with firewall functions 26,28. The firewall functions are configured with the rules shown in Table 1 to allow real-time communications such as those based on H.323. The rules take into account the two or more new well known ports proposed under an earlier invention, referred to as X,Y and Z. Port Z may in practice be equal to either X or Y.

Table 1:

Rule	From IP Address	From Port	To IP Address	To Port	IP protocol	Application
1	Any	Any	External server	Z	TCP	Outbound Multiplex Connection
2	External server	Z	Any	Any	TCP	Inbound Multiplex Connection
3	Any	Any	External server	X	UDP	Outbound Media (RTP)
4	External server	X	Any	Any	UDP	Inbound Media (RTP)
5	Any	Any	External server	Y	UDP	Outbound Media (RTCP)
6	External server	Y	Any	Any	UDP	Inbound Media (RTCP)

In Table 1, ideally the listed port numbers, X, Y and Z are registered port numbers according to standards agreed to by IANA. The advantage of these ports being industry standard ports is that intermediary equipment such as firewalls and routers would know the associated media is real-time traffic and could, therefore, handle it appropriately, for example a router could give it higher priority forwarding in order to minimise delays.

In order for H.323 terminals 10 in the first enterprise 2 to communicate with other H.323 terminals 12 in the second enterprise 4, there must exist a shared network 20 to which a external server 40 is connected, for example, via a router 38. The external server 40 has a public IP address 44, for example 45.6.7.8. The external server would also have new well known ports numbers X,Y and Z 46 that would have to be agreed and registered in advance with IANA.

Figure 3 shows the communications paths between the various entities from the perspective of the first terminal 10, the first proxy interface 11, the first firewall 26, the first NAPT router 32 and the external server 40. The figure shows the multiplex connection 51 between the proxy interface agent 11 and the external server 40, via the firewall 26 and NAPT router 32. Within the multiplex connection 51 are one or more logical channels 52, 53. One of these is the control channel 52, while the others 53 carry signalling protocols such as H.225 RAS, H.225 call signalling, H.245, SIP and MGCP. As part of the operation described below, the proxy interface agent 11 will send probe packets 55 to the external server 40, and establish UDP connections 56, 57 between the terminal 10 and the external server 40. One or more logical channels may be multiplexed into the UDP connections 56, 57 to carry media such as RTP and RTCP for example.

The proxy interface agent 11 may operate in one of a number of modes depending on operational requirements. Principally it can be either protocol agnostic or protocol aware. If it is protocol agnostic the external server 40

will command the proxy interface agent 11 to open and close any UDP sockets needed. This is the most flexible mode as it allows terminals employing new protocols to be added to the private network without upgrading the proxy interface agents 11. However, without due care, this could present a security threat as third parties could instruct the proxy interface agent to open UDP channels for illicit purposes. For this reason, if this mode is adopted, it is recommended that as a minimum the proxy interface agent 11 perform some form of auditing. If the proxy interface agent 11 is protocol aware, then it can allocate ports when instructed by the external server 40, but not implement the relaying function until it has observed appropriate protocol signalling to indicate that these ports are being used for an approved application. This mode is more secure, but less flexible with regard to deploying new applications, or application upgrades. For simplicity, the example described below assumes the proxy interface agent 11 is operating in the protocol agnostic mode.

When the proxy interface agent 11 is enabled it establishes a multiplex connection 51 as a communications channels to the external server 40 by initiating an outbound TCP connection to the address and port of the external server 44, 46. (This connection will typically be authenticated and encrypted, but such matters are beyond the scope of this document.)

The multiplex connection 51 is capable of transporting the information pertaining to multiple TCP and UDP sessions 52, 53. Some of the logical channels within the multiplex connection 51 will be statically allocated; in particular

the control channel 52. Other logical channels can be dynamically created as the need arises. Some of the logical channels 53 will be relayed to/from the terminal 10 by the proxy interface agent 11. With each such logical channel the proxy interface agent 11 (or the external server depending on implementation) associates the IP addresses and ports of the specific TCP or UDP connection used between the proxy interface agent 11 and the terminal 10. In other words, the proxy interface agent makes an association between a transport address of the terminal and the transport address on the its own end of a logical channel.

As part of the initial configuration, the external server 40 may instruct the proxy interface agent 11 to create sockets to listen for registration information and outgoing call attempts from the terminal 10.

If the terminal 10 subsequently attempts to register with a gatekeeper/server, such messages (H.225 RAS, SIP REGISTER etc.) may be sent to the proxy interface agent 11. The proxy interface agent 11 will forward the registration messages to the external server 40 via a logical channel 52 or 53. Any responses are sent using the reverse route. The external server 40 will store the terminal's private transport address 14 along with the identity or transport address of the multiplex connection 51 on which the registration was received. This information is sufficient to forward incoming calls to the terminal when the need arises.

To establish an incoming call the external server 40 needs to establish a call control channel (H.225 call control

for H.323 or SIP) to the terminal 10 via the proxy interface agent 11. If an appropriate logical channel 53 does not already exist between the external server 40 and the proxy interface agent 11, such a logical channel is instantiated. As part of this process, the terminal's private transport address (IP address and port) 14 to which the proxy interface agent 11 is to create the TCP or UDP connection 54 is specified. The messages needed to create the logical channel 53 are exchanged between the external server 40 and the proxy interface agent 11 using the control logical channel 52.

Once the logical channel for the call control signalling has been created the external server 40 can send an H.323/SIP create call message (Setup for H.323, INVITE for SIP etc) to the proxy interface agent 11. The proxy interface agent will then relay this message to the terminal 10 using the TCP or UDP connection 54 established when the logical channel 53 was created.

In the case of H.323 it may be necessary to establish an H.245 connection between the external server 40 and the terminal 10. The address within the terminal 10 to which this connection is to connect is contained in the responses sent back to the external server 40 by the terminal 10. If the external server 40 chooses to establish such an H.245 session, then it creates a new logical channel 53 in the same way it created the call-signalling channel. As part of this procedure the proxy interface agent 11 will establish a TCP connection to the private IP address and port specified in the terminal's responses.

For an outgoing call, a signalling path can be created between the terminal 10 and the external server 40 when the terminal 10 connects and sends a create call message (Setup for H.323, INVITE for SIP etc) to the proxy interface agent 11. If a logical channel 53 for this type of connection does not already exist within the multiplex connection 51, then such a logical channel is created by the proxy interface agent 11 using the control channel 52. The proxy interface agent 11 can then relay the message(s) to the external server 40.

If a separate H.245 connection is required for the outgoing call, the external server 40 will create a new logical channel 53 within the multiplex connection 51 and instruct the proxy interface agent 11 to create a listening socket. The values of address and port of the created socket are returned to the external server 40, which it includes in the H.323 signalling sent in response to the Setup message. This information allows the terminal 10 to connect to the listening socket created by the proxy interface agent 11.

Once the necessary incoming or outgoing call control paths have been established it may be necessary to establish outbound and inbound media paths. As described earlier, the media paths of all currently defined IP based multimedia applications (including H.323, SIP and MGCP) use RTP. RTP is based on UDP, and a unidirectional RTP connection requires both forward and reverse UDP paths to be established. It is, therefore, necessary to establish UDP paths from the terminal 10 to the external server 40 via the proxy interface agent 11, and from the external server 40 to the terminal 10, again via the proxy

interface agent 11. Additionally, the RTP and RTCP connections require a fixed relationship between the ports they use. Therefore, in addition to being able to open a single port at a time, it is necessary to be able to open UDP port pairs which have the necessary RTP/RTCP port number relationship. Therefore, while the text below describes opening a single connection, the same principles can be employed to simultaneously request and open port pairs.

The following discussion assumes that the H.323 protocol is being used. The sequences of protocol messages versus control messages may vary for other protocols (such as SIP and MGCP), but the principles remain the same.

To establish a UDP path between the terminal 10 and the external server 40, the external server 40 instructs the proxy interface agent 11 to open a UDP port (or port pair) that the terminal 10 can connect to. The external server 40 also specifies a token that the proxy interface agent 11 should associate with the connection.

On successfully opening the port, the proxy interface agent 11 indicates to the external server 40 the identity of the port. The external server is then able to issue the necessary signalling commands to open a media channel (e.g. H.245 Open Logical Channel in the case of H.323) containing the private IP address and port on the proxy interface agent 11 to which the terminal 10 should send its UDP data. On reception of this command, the proxy interface agent 11 relays the command to the terminal 10 using the connection established previously for this purpose.

The terminal 10 can now start sending RTP and RTCP UDP packets 56 to the proxy interface agent 11. However, prior to forwarding these packets to the external server 40, the proxy interface agent 11 must send probe packets 55 which contain the token specified by the external server 40 when the connection was initially configured. In addition to creating a private-to-public address mapping in the NAPT, the presence of the token allows the external server 40 to associate UDP packets 57 received from the source of these probe packets 55 with the correct logical media channel. Note that it is preferable to defer sending the probe packets 55 for as long as possible as if they are sent too early the address mappings created in the NAT may time out before any media data 56 is sent. Also, it is necessary to be aware that, being UDP, the probe packets 55 may be lost. It is therefore necessary to have the ability to send more than one probe packet 55 for a given connection. Once a probe packet 55 has been sent, the proxy interface agent can relay received UDP data 56 to the external server 40 (as item 57). Alternatively, the token information can be multiplexed into each UDP packet that is sent. Additionally, multiple logical channels may be multiplexed onto one or more UDP connections.

The method of operation is similar for an inbound UDP connection. The external server 40 instructs the proxy interface agent 11 to open a port (or port pair) that can be used to send UDP data to the terminal 10. The proxy interface agent 11 informs the external server 40 of the identity of this port. The external server 40 can then include this information in the protocol specific signalling command to open a media channel (e.g. H.245

Open Logical Channel in the case of H.323) that is sent to the terminal 10 via the proxy interface agent 11. The terminal 10 will reply to this command, giving the private IP address and port at which it wishes to receive UDP data for the connection. This message is relayed back to the external server 40. The external server 40 can then inform the proxy interface agent 11 of the address to which it should relay UDP data for this connection. Further, to create the public-to-private address mapping in the NAT, the external server 40 requests that the proxy interface agent 11 send probe packets 55 for this connection to the external server 40 containing a token. This creates a private-to-public address mapping that in turn acts as a public-to-private address mapping for data sent in the reverse direction. The external server 40 uses the token in the probe packet 55 to determine which NAT address and port it should send UDP data to for this session 57. The external server 40 may now start sending UDP media 57 to this address. The NAT will relay this to the proxy interface agent 11, which will in turn relay it to the terminal 10 (as item 56), thus completing the connection.

When the UDP connections are no longer required, the external server 40 will instruct the proxy interface agent 11 to close the associated sockets. Any private-to-public address mappings in the NAPT will eventually time out as no data will be passing through them.

In this illustration of the invention, we have assumed that the external server is a single device with a single IP address. In other embodiments of the invention the 'external server' may be several co-operating devices. Additionally, the external server device(s) may each have

one or multiple IP addresses. Where multiple IP addresses are used, the normal practise is to allocate them from a single subnet, then the programming of the firewall rules becomes specifying the allowed ports to and from a subnet rather than individual IP addresses.

Note that the private IP address and port numbers of an H.323 terminal may in fact be the same as the public IP address and port numbers to which it is mapped, in which case the mapping is transparent.

The advantages of the approach described above are that:

- NAT and firewall functions do not need to be upgraded.
- Latency of the signal is kept to a minimum.
- Organisation only require a protocol agnostic proxy interface agent(s) that can be used with any appropriate real-time protocol.
- The IP address(es) of the enterprise does not become public knowledge through process of making calls with that enterprise
- Quality of service and other usage based policies (bandwidth utilisation for example) may be implemented piecemeal and don't need a single consistent end-to-end solution. For example, the external server may instruct the proxy interface agent to process one media stream within a call with a certain QOS level, using a method that is appropriate to the connection between the proxy interface agent and the external server, the external server may then map that to corresponding QOS levels available to it in the core network. Likewise, a method of encryption may be used between the proxy interface agent and the external server independently of security

mechanisms used for the other parts (legs) of the call.

In summary, the invention provides a method and a system for allowing H.323 (or other real-time protocol conformant endpoints) terminals located in private IP networks that: does not compromise the existing security procedures and measures; that avoids the need to upgrade existing firewalls, routers and proxies; and that allows full NAT to be applied to IP connections without the NAT function interpreting or understanding the communications protocol being used. The invention also permits standard H.323 equipment in one private network to communicate with other H.323 terminals in the same or different private and/or public IP networks via an protocol independent proxy interface agent and via an H.323 proxy server using a shared or public IP network.

Organisations can therefore subscribe to a shared resource in a shared IP network. Costs are kept to a minimum and security is not compromised.

Claims

1. A communications system for handling a communications session, comprising a first local terminal, an external server, one or more logical channels between the first local terminal and the external server for carrying the communications session over a shared communications network, said communication means including a first NAT function through which the communications session must pass, in which:

i) the first local terminal has at least one transport address for the communications session;

ii) the first NAT function applies network address mappings on the transport addresses on connections between the first terminal and the shared communications network;

iii) the system includes a first proxy interface agent arranged to act on behalf of the first local terminal in communications with the external server;

iv) the first proxy interface agent is capable of establishing a logical channel on an outbound connection to the external server, said logical channel serving as a control channel between the first proxy interface agent and the external server;

characterised in that:

v) said outbound connections are dynamic outbound connections established by the first proxy interface agent;

vi) the external server is adapted to request the first proxy interface agent to make association(s) between the transport address(es) of the first local terminal and identifiable logical channel(s) between the first proxy interface agent and the external server, said identifiable logical channel(s) being established on one or more of said dynamic outbound connections from the first proxy interface agent to the external server.

2. A method of handling a communications session in a communications system, the communications system comprising a first local terminal, an external server, a first proxy interface agent between the first local terminal and the shared network, said communication means including a first NAT function through which the communications session must pass, in which the method comprises the steps of:

i) carrying the communications session over a shared communications network over one or more logical channels between the first local terminal and the external server, the first local terminal having at least one transport address for the communications session;

ii) allowing the first NAT function to continue to apply network address mappings on the transport addresses on connections between the first terminal and the shared communications network;

iii) using the first proxy interface agent to act on behalf of the first local terminal in communications with the external server;

iv) using the first proxy interface agent to establish a logical channel on an outbound connection to the external server, said logical channel serving as a control channel between the first proxy interface agent and the external server;

characterised in that the method comprises the steps of:

v) using the first proxy interface agent to establish dynamic outbound connections to the external server;

vi) in response to a request from the external server directed to the first proxy interface agent, using the first proxy interface agent to make one or more associations between the transport address(es) of the first local terminal and identifiable logical channel(s) between the first proxy interface agent and the external server, said identifiable logical channel(s) being established on one or more of said dynamic outbound connections from the first proxy interface agent to the external server.

3. A method as claimed in Claim 2, in which the transport address(es) of the first local terminal are assigned dynamically.

4. A method as claimed in Claim 2 or Claim 3, in which the transport address(es) of the external server are assigned dynamically.

5. A method as claimed in Claim 2 or Claim 3, in which none of the transport address(es) of the external server

are assigned dynamically.

6. A method as claimed in any of Claims 2 to 5, in which the communications system includes a first firewall through which the communications session must pass, the first firewall being configured to restrict certain types of communication between the first local terminal and the shared communications network and being configured not to restrict communication between the first proxy interface agent and the external server.

7. A method as claimed in Claim 6, in which at least one of the transport address(es) of the external server have at least one pre-assigned port, and the outbound connections from the first proxy interface agent to the external server uses said pre-assigned port(s).

8. A method as claimed in Claim 7, in which all the transport address(es) of the external server have pre-assigned ports.

9. A method as claimed in Claim 8, in which all the transport address(es) of the external server have at most two pre-assigned ports.

10. A method as claimed in any of Claims 2 to 9, in which all the transport address(es) of the proxy interface agent are assigned dynamically.

11. A method as claimed in any of Claims 2 to 9, in which at least one of the transport address(es) of the proxy interface agent uses pre-assigned ports.

12. A method as claimed in any of Claims 2 to 9, in which all the transport address(es) of the proxy interface agent uses pre-assigned ports.

13. A method as claimed in any of Claims 2 to 12, in which the communications system includes a second local terminal and the external server is a proxy server between the first terminal and the second terminal that acts for each terminal as a proxy for the other terminal during the course of the communications session.

14. A method as claimed in any of Claims 2 to 13, in which the shared communications network includes the public communications network.

15. A method as claimed in any of Claims 2 to 14, in which the shared communications network includes the Internet.

16. A method as claimed in any of Claims 2 to 15, in which the proxy interface agent is co-located with the local terminal.

17. A method as claimed in any of Claims 2 to 15, in which the proxy interface agent is remote from the local terminal.

18. A method as claimed in any of Claims 2 to 17, in which there is more than one local terminal for the proxy interface agent.

19. A method as claimed of 18, in which the proxy interface agent simultaneously acts on behalf of terminals

using different real-time protocols.

20. A communications system substantially as herein described, with reference to or as shown in the accompanying drawings.

21. A method of handling a communications session in a communications system substantially as herein described, with reference to or as shown in the accompanying drawings.

The Patent Office

46

INVESTOR IN PEOPLE

Application No:	GB 0029179.9	**Examiner:**	Jared Stokes
Claims searched:	1 to 19	**Date of search:**	17 September 2001

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.S): H4K (KTKX)

Int Cl (Ed.7): H04L (29/12)

Other: On-Line - EPODOC, JAPIO, WPI

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
A,P	WO 01/30036 A1 (Ericsson) See abstract	-
A	US 6 058 431 (Srisuresh) See abstract	-

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
		E	Patent document published on or after, but with priority date earlier than, the filing date of this application.
&	Member of the same patent family		